Objective:

To create a software pipeline to identify lane boundaries in a video from a front facing camera on a car. This assignment was already done as a part of our college Self Driving Car project.

Algorithm:

- Simple Lane Detection:
    1. Convert RGB image to grayscale image
    2. Convert RGB space to HSV space and identify yellow colour
    3. Apply Gaussian blur to remove noise
    4. Use canny edge detection to get the edges in the image
    5. A polygon is chose as the region of interest

```
lower_left = [imshape[1]/9,imshape[0]]
lower_right = [imshape[1]-imshape[1]/9,imshape[0]]
top_left = [imshape[1]/2-imshape[1]/8,imshape[0]/2+imshape[0]/10]
top_right = [imshape[1]/2+imshape[1]/8,imshape[0]/2+imshape[0]/10]
```

6. Used hough lines in the region of interest and merged the resultant lines with the input image

- Advanced Lane Detection:

    Following are the steps to be followed:

    1. Compute the camera calibration matrix and distortion coefficients given a set of chessboard images.
    2. Apply distortion correction to raw images.
    3. Use color transform and gradients to create a thresholded binary image.
    4. Apply a perspective transform (birds-eye view) to rectify binary image.
    5. Detect lane pixels and fit a polynomial expression to find the lane boundary.
    6. Determine the curvature of the lane and vehicle position with respect to the center.
    7. Overlay the detected lane boundaries onto the original image.

Dependencies and My environment:

1. Python3
2. Jupyter Notebook, Numpy, OpenCV 3.0
3. Matplotlib, glob, pickle, scikit-image, IPython

How to compile and run:

1. To see **advanced lane detection** results, open terminal in the lane detection folder and run: **jupyter notebook**; On the GUI that pops up, Execute the command: **Cell->Run all.** The input video is taken from **input_videos** folder and the processed video is saved in the **output_videos** folder in the name **project_lane.mp4**.
   NOTE: If you get a pickle protocol error in running the notebook, then open python3 interpreter in the lane_detection folder and import calibration_main.py and run its function: do_calibration() once.

2. To see **simple lane detection** results, run the script

   python simple_lane_detection.py -v input_videos/white.mp4

3. The input video is taken from **input_videos** folder and the processed video is saved in the **output_videos** folder in the name **simple_white.mp4**.